

Amendments to the Claims

Please amend claims 1, 5, 6, and 10-12. Please cancel claim 16. The currently pending claims after amendment are listed below.

1. (Currently Amended) A method for database query optimization in a computer system having a plurality of central processors, comprising the steps of:
- defining a plurality of logical partitions of said computer system, each logical partition having a respective processor resource assignment, wherein each task executing in said computer system is assigned to a respective one of said logical partitions and wherein the definition of a plurality of logical partitions may be dynamically altered;
 - defining a database query;
 - constructing a first search strategy for said database query, said first search strategy being dependent on a first processor resource assignment at the time said step of constructing a first search strategy is performed;
 - invoking said database query for execution in a first logical partition, said invoking step being performed after said step of constructing a first search strategy;
 - automatically comparing a second processor resource assignment to said first processor resource assignment, said second processor resource assignment being associated with said first logical partition at the time said invoking said database query for execution step is performed; and
 - automatically constructing a second search strategy dependent on said second processor resource assignment, said step of automatically constructing a second search strategy being performed dependent on the results of said comparing step.
2. (Original) The method for database query optimization of claim 1, wherein said respective processor resource assignment of each partition comprises a respective number of virtual processors of each partition, said respective number being an integer.

1 3. (Original) The method for database query optimization of claim 2, wherein said step of
2 defining a plurality of logical partitions comprises defining at least one set of processors which is
3 shared by a set of said logical partitions, said set of said logical partitions containing at least two
4 partitions, said respective processor resource assignment of each partition of said set of partitions
5 including said set of processors.

1 4. (Original) The method for database query optimization of claim 1, further comprising the
2 step of:
3 saving said first search strategy in a persistent object for later execution, said saving step
4 including saving said first processor resource assignment in said object.

1 5. (Currently Amended) ~~The A~~ method for database query optimization ~~of claim 4 in a~~
2 computer system having a plurality of central processors, further comprising the steps of:
3 defining a plurality of logical partitions of said computer system, each logical partition
4 having a respective processor resource assignment, wherein each task executing in said computer
5 system is assigned to a respective one of said logical partitions and wherein the definition of a
6 plurality of logical partitions may be dynamically altered;
7 defining a database query;
8 constructing a first search strategy for said database query, said first search strategy being
9 dependent on a first processor resource assignment at the time said step of constructing a first
10 search strategy is performed;
11 invoking said database query for execution in a first logical partition, said invoking step
12 being performed after said step of constructing a first search strategy;
13 comparing a second processor resource assignment to said first processor resource
14 assignment, said second processor resource assignment being associated with said first logical
15 partition at the time said invoking said database query for execution step is performed;

16 automatically constructing a second search strategy dependent on said second processor
17 resource assignment, said step of automatically constructing a second search strategy being
18 performed dependent on the results of said comparing step;
19 saving said first search strategy in a persistent object for later execution, said saving step
20 including saving said first processor resource assignment in said object;
21 invoking a previously saved search strategy for execution in a second logical partition, said
22 second logical partition being different from said first logical partition;
23 identifying a third processor resource assignment associated with said second logical
24 partition;
25 comparing said third processor resource assignment to said first processor resource
26 assignment; and
27 automatically constructing a third search strategy for execution of said database query
28 depending on the results of said comparing step.

1 6. (Currently Amended) ~~The A~~ method for database query optimization ~~of claim 1~~ in a
2 computer system having a plurality of central processors, further comprising the steps of:
3 defining a plurality of logical partitions of said computer system, each logical partition
4 having a respective processor resource assignment, wherein each task executing in said computer
5 system is assigned to a respective one of said logical partitions and wherein the definition of a
6 plurality of logical partitions may be dynamically altered;
7 defining a database query;
8 constructing a first search strategy for said database query, said first search strategy being
9 dependent on a first processor resource assignment at the time said step of constructing a first
10 search strategy is performed;
11 invoking said database query for execution in a first logical partition, said invoking step
12 being performed after said step of constructing a first search strategy;
13 comparing a second processor resource assignment to said first processor resource
14 assignment, said second processor resource assignment being associated with said first logical
15 partition at the time said invoking said database query for execution step is performed;
16 determining whether a user has specified that automatic construction of another search
17 strategy be disabled;
18 automatically constructing a second search strategy dependent on said second processor
19 resource assignment, said step of automatically constructing a second search strategy being
20 performed dependent on the results of said comparing step, wherein said step of automatically
21 constructing a second search strategy dependent on said second processor resource assignment is
22 performed only if said determining step determines that a user has not specified that automatic
23 construction of another search strategy be disabled.

1 7. (Original) A program product for database query optimization in a computer system
2 having a plurality of central processors and a dynamic logical partitioning mechanism, said
3 dynamic logical partitioning mechanism supporting a plurality of defined logical partitions of said
4 computer system, each logical partition having a respective processor resource assignment,
5 wherein each task executing in said computer system is assigned to a respective one of said
6 logical partitions and wherein the definition of a plurality of logical partitions may be dynamically
7 altered, said program product comprising a plurality of processor executable instructions recorded
8 on signal-bearing media, wherein said instructions, when executed by at least one central
9 processor of said computer system, cause the system to perform the steps of:

10 receiving a definition of a database query;

11 constructing a first search strategy for said database query, said first search strategy being
12 dependent on a first processor resource assignment at the time said step of constructing a first
13 search strategy is performed;

14 invoking said database query for execution in a first logical partition, said invoking step
15 being performed after said step of constructing a first search strategy;

16 comparing a second processor resource assignment to said first processor resource
17 assignment, said second processor resource assignment being associated with said first logical
18 partition at the time said invoking said database query for execution step is performed; and

19 automatically constructing a second search strategy dependent on said second processor
20 resource assignment, said step of automatically constructing a second search strategy being
21 performed dependent on the results of said comparing step.

1 8. (Original) The program product for database query optimization of claim 7, wherein said
2 respective processor resource assignment of each partition comprises a respective number of
3 virtual processors of each partition, said respective number being an integer.

1 9. (Original) The program product for database query optimization of claim 7, wherein said
2 instructions further cause said computer system to perform the step of:

3 saving said first search strategy in a persistent object for later execution, said saving step
4 including saving said first processor resource assignment in said object.

1 10. (Currently Amended) ~~The A~~ program product for database query optimization of claim 9
2 in a computer system having a plurality of central processors and a dynamic logical partitioning
3 mechanism, said dynamic logical partitioning mechanism supporting a plurality of defined logical
4 partitions of said computer system, each logical partition having a respective processor resource
5 assignment, wherein each task executing in said computer system is assigned to a respective one
6 of said logical partitions and wherein the definition of a plurality of logical partitions may be
7 dynamically altered, said program product comprising a plurality of processor executable
8 instructions recorded on signal-bearing media, wherein said instructions, when executed by at
9 least one central processor of said computer system, further cause said computer system to
10 perform the steps of:

11 receiving a definition of a database query;

12 constructing a first search strategy for said database query, said first search strategy being
13 dependent on a first processor resource assignment at the time said step of constructing a first
14 search strategy is performed;

15 invoking said database query for execution in a first logical partition, said invoking step
16 being performed after said step of constructing a first search strategy;

17 comparing a second processor resource assignment to said first processor resource
18 assignment, said second processor resource assignment being associated with said first logical
19 partition at the time said invoking said database query for execution step is performed;

20 automatically constructing a second search strategy dependent on said second processor
21 resource assignment, said step of automatically constructing a second search strategy being
22 performed dependent on the results of said comparing step;

23 saving said first search strategy in a persistent object for later execution, said saving step
24 including saving said first processor resource assignment in said object;
25 invoking a previously saved search strategy for execution in a second logical partition, said
26 second logical partition being different from said first logical partition;
27 identifying a third processor resource assignment associated with said second logical
28 partition;
29 comparing said third processor resource assignment to said first processor resource
30 assignment; and
31 automatically constructing a third search strategy for execution of said database query
32 depending on the results of said comparing step.

11. (Currently Amended) ~~The A~~ program product for database query optimization ~~of claim 7~~
in a computer system having a plurality of central processors and a dynamic logical partitioning
mechanism, said dynamic logical partitioning mechanism supporting a plurality of defined logical
partitions of said computer system, each logical partition having a respective processor resource
assignment, wherein each task executing in said computer system is assigned to a respective one
of said logical partitions and wherein the definition of a plurality of logical partitions may be
dynamically altered, said program product comprising a plurality of processor executable
instructions recorded on signal-bearing media, wherein said instructions, when executed by at
least one central processor of said computer system, further cause said computer system to
perform the steps of:

- receiving a definition of a database query;
- constructing a first search strategy for said database query, said first search strategy being
dependent on a first processor resource assignment at the time said step of constructing a first
search strategy is performed;
- invoking said database query for execution in a first logical partition, said invoking step
being performed after said step of constructing a first search strategy;
- comparing a second processor resource assignment to said first processor resource
assignment, said second processor resource assignment being associated with said first logical
partition at the time said invoking said database query for execution step is performed;
- determining whether a user has specified that automatic construction of another search
strategy be disabled;
- automatically constructing a second search strategy dependent on said second processor
resource assignment, said step of automatically constructing a second search strategy being
performed dependent on the results of said comparing step, wherein said step of automatically
constructing a second search strategy dependent on said second processor resource assignment is
performed only if said determining step determines that a user has not specified that automatic
construction of another search strategy be disabled.

28 12. (Currently Amended) A computer system, comprising:
29 a plurality of central processing units;
30 a memory;
31 a logical partitioning mechanism supporting a plurality of defined logical partitions of said
32 computer system, each logical partition having a respective processor resource assignment,
33 wherein each task executing in said computer system is assigned to a respective one of said
34 logical partitions and wherein the definition of said logical partitions may be dynamically altered;
35 a database;
36 a database management system for managing said database, wherein said database
37 management system:
38 (a) performs query optimization of a database query for said database to produce a
39 first search strategy, said first search strategy being dependent on a first processor resource
40 assignment;
41 (b) responsive to invoking said first query search strategy for execution, compares
42 said first processor resource assignment with a second processor resource assignment
43 associated with a logical partition of execution at the time said first search strategy is
44 invoked for execution; and
45 (c) depending on the results of said comparison performed in (b), automatically
46 constructs a second search strategy dependent on said second processor resource
47 assignment;
48 wherein said database management system further determines whether a user has specified that (c)
49 be disabled, and disables (c) responsive to determining that a user has so specified.

1 13. (Original) The computer system of claim 12, wherein said respective processor resource
2 assignment of each partition comprises a respective number of virtual processors of each partition,
3 said respective number being an integer.

1 14. (Original) The computer system of claim 13, wherein said logical partitioning mechanism
2 supports the definition of at least one set of processors which is shared by a set of said logical
3 partitions, said set of said logical partitions containing at least two partitions, said respective
4 processor resource assignment of each partition of said set of partitions including said set of
5 processors.

1 15. (Original) The computer system of claim 12, wherein said database management system
2 saves said first search strategy in a persistent object for later execution, said persistent object
3 including said first processor resource assignment.

16. (Cancelled)